J 6

```
MMM         MMM  TTTTTTTTTTTTTTT     AAAAAAAAA        AAAAAAAAA        CCCCCCCCCCCC  PPPPPPPPPPPP
MMM         MMM  TTTTTTTTTTTTTTT     AAAAAAAAA        AAAAAAAAA        CCCCCCCCCCCC  PPPPPPPPPPPP
MMM         MMM  TTTTTTTTTTTTTTT     AAAAAAAAA        AAAAAAAAA        CCCCCCCCCCCC  PPPPPPPPPPPP
MMMMMM   MMMMMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMMMMM   MMMMMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMMMMM   MMMMMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMM   MMM   MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMM   MMM   MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMM   MMM   MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPPPPPPPPPPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPPPPPPPPPPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPPPPPPPPPPP
MMM         MMM        TTT        AAAAAAAAAAAAAAA  AAAAAAAAAAAAAAA  CCC              PPP
MMM         MMM        TTT        AAAAAAAAAAAAAAA  AAAAAAAAAAAAAAA  CCC              PPP
MMM         MMM        TTT        AAAAAAAAAAAAAAA  AAAAAAAAAAAAAAA  CCC              PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA     CCCCCCCCCCCC  PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA     CCCCCCCCCCCC  PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA     CCCCCCCCCCCC  PPP
```

```
**FILE**ID**ACPCTR
```

```
  AAAAAA      CCCCCCCC  PPPPPPPP      CCCCCCCC  TTTTTTTTTT  RRRRRRRR
  AAAAAA      CCCCCCCC  PPPPPPPP      CCCCCCCC  TTTTTTTTTT  RRRRRRRR
AA      AA  CC          PP      PP  CC               TT       RR      RR
AA      AA  CC          PP      PP  CC               TT       RR      RR
AA      AA  CC          PP      PP  CC               TT       RR      RR
AA      AA  CC          PPPPPPPP    CC               TT       RRRRRRRR
AA      AA  CC          PPPPPPPP    CC               TT       RRRRRRRR
AAAAAAAAAA  CC          PP          CC               TT       RR  RR
AAAAAAAAAA  CC          PP          CC               TT       RR  RR
AA      AA  CC          PP          CC               TT       RR    RR
AA      AA  CC          PP          CC               TT       RR    RR        ....
AA      AA    CCCCCCCC  PP            CCCCCCCC       TT       RR      RR      ....
AA      AA    CCCCCCCC  PP            CCCCCCCC       TT       RR      RR      ....
                                                                              ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL               II     SS
LL               II     SS
LL               II     SS
LL               II     SS
LL               II       SSSSSS
LL               II       SSSSSS
LL               II             SS
LL               II             SS
LL               II             SS
LL               II             SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
0
0 MODULE ACPCTR (LANGUAGE (BLISS32) ,
0               IDENT = 'V04-000'
0               ) =
1 BEGIN
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER  *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY  *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE  *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT  *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1
1 !++
1 !
1 ! FACILITY:  MTAACP
1 !
1 ! ABSTRACT:
1 !       This module handles acp control functions.
1 !
1 ! ENVIRONMENT:
1 !
1 !       Starlet operating system, including privileged system services
1 !       and internal exec routines.
1 !
1 !--
1
1 !
1 !
1 ! AUTHOR: D. H. Gillespie,      CREATION DATE:  09-JUL-1977
1 !
1 ! MODIFIED BY:
1 !
1 !       V03-005 MMD0236         Meg Dumont,     4-Feb-1984  15:13
1 !               Add support for FIB$C_CLSEREXCP when set with IO$_ACPCONTROL.
1 !
1 !       V03-004 MMD0171         Meg Dumont,     9-May-1983  15:12
1 !               Fix to make USER_STATUS defined consistently within module
1 !
1 !       V03-003 MMD0149         Meg Dumont,     26-Apr-1983  8:51
1 !               Change references to 80 to the symbol ANSI_LBLSZ
```

```
1 !
1 !       V03-002 MMD0001         Meg Dumont,     3-Jan-1983  15:22
1 !               Added a call to stop access to a file if the trailer labels
1 !               have been read.
1 !
1 !       V03-001 STJ0309         Steven T. Jeffreys       1-Jun-1982
1 !               Added handler for REMOUNT control function.  It's a NOP.
1 !
1 !       V02-011 DMW00077        David Michael Walp      8-Feb-1982
1 !               Stored account and user name during mount time
1 !
1 !       V02-010 DMW00034        David Michael Walp      15-Sep-1981
1 !               Fixed Cancel I/O vs Dismount race condition
1 !
1 !       V02-009 DMW00024        David Michael Walp      20-Jul-1981
1 !               Change to RET_FREE_PAGE to not contract region P0 every time
1 !               space is returned.
1 !
1 !       V02-008 DMW00009        David Michael Walp      14-Mar-1981
1 !               Changed calculation of CCB address
1 !
1 !       V02-007 DMW00001        David Michael Walp      11-Nov-1980
1 !               New BLISS compiler, FUNCTION declaration changed from
1 !               BBLOCK to BLOCK. Old compiler used to give a longword
1 !               with a declaration of "BBLOCK [1]".
1 !
1 !       V02-006 REFORMAT        Maria del C. Nasr       30-Jun-1980
1 !
1 !       V02-005 MCN0017         Maria del C. Nasr       18-Jun-1980
1 !               Add a call to START_VIO after completing the next volume
1 !               write.  This is part of the fix for multivolume processing,
1 !               in which a new volume should be requested when the EOT is
1 !               sensed in writing the header labels, and not wait for the
1 !               data to be written.
1 !
1 !       V02-004 SPR27361        Maria del C. Nasr       10-Jun-1980
1 !               Add a call to START_VIO after completing the next volume
1 !               read.  This is part of the general fix which delays IO
1 !               posting until all IO is successfully completed.
1 !
1 !       A0103   MCN0007         Maria del C. Nasr       13-Nov-1979   19:35
1 !               Set single directory structured device bit (DEV$M_SDI)
1 !
1 !
1 !**
1
1 LIBRARY 'SYS$LIBRARY:LIB.L32';
1
1 REQUIRE 'SRC$:MTADEF.B32';
1
1 FORWARD ROUTINE
1     MTA_ACPCNTRL          : NOPRES NOVALUE,  ! control function dispatch
1     MTA_MOUNT     : NOPRES NOVALUE,          ! mount function
1     CANCEL_IO     : COMMON_CALL NOVALUE,     ! cancel i/o control
1     DO_CANCEL     : COMMON_CALL NOVALUE,     ! do actual cancelation of io
1     MOUNT         : COMMON_CALL NOVALUE,     ! mount control function, kernel mode
1     STALL         : COMMON_CALL NOVALUE,     ! stall cancel
```

```
    TERMINATE_VOL: COMMON_CALL NOVALUE;   ! terminate mount volume request

EXTERNAL ROUTINE
    CANCEL_OP_REPLY     : COMMON_CALL,               ! cancel reply from operator
    IO_DONE,                                         ! complete io
    NEXT_VOL_READ       : L$NEXT_VOL_READ NOVALUE, ! get next vol for read
    NEXT_VOL_WRITE      : L$NEXT_VOL_WRIT NOVALUE, ! get next vol for write
    READ_BLOCK  : COMMON_CALL,                       ! read a tape block
    RET_FREE_PAGE       : COMMON_CALL,     ! return virtual page to free list
    RETURN_ACL_ERR      : COMMON_CALL,     ! return blocked virtual io in error
    SEND_ERRLOG,
    SPACE_TM    : COMMON_CALL,                       ! space tape mark
    START_VIO   : COMMON_CALL,                       ! start up virtual io
    STOP_VIO    : COMMON_CALL,                       ! Disallow VIO's
    SYS$QIOW    : ADDRESSING_MODE (ABSOLUTE),
    ZERO_CHANNEL        : COMMON_CALL;               ! zero channel

EXTERNAL
    SCH$GL_PCBVEC       : REF VECTOR ADDRESSING_MODE (ABSOLUTE),
    CURRENT_UCB : REF BBLOCK,
    CURRENT_WCB : REF BBLOCK,   ! address of current window control block
    HDR1        : REF BBLOCK,   ! hdr1(eof1) label
    IO_CHANNEL,
    IO_PACKET   : REF BBLOCK,   ! address of current io packet
    USER_STATUS : VECTOR [2];   ! address of user status
```

```
GLOBAL ROUTINE MTA_ACPCNTRL : NOPRES NOVALUE =

!++
!
! FUNCTIONAL DESCRIPTION:
!       This routine handles the acp control function.
!
! CALLING SEQUENCE:
!       MTA_ACPCNTRL()
!
! INPUT PARAMETERS:
!       None
!
! IMPLICIT INPUTS:
!       CURRENT_UCB - address of current unit control block
!       CURRENT_VCB - address of current volume control block
!       IO_PACKET - address of current io request packet
!       QUEUE_HEAD  - address of acp queue
!
! OUTPUT PARAMETERS:
!       None
!
! IMPLICIT OUTPUTS:
!       LOCAL_FIB - copy of user's fib
!
! ROUTINE VALUE:
!       None
!
! SIDE EFFECTS:
!       None
!
!--

    BEGIN

    EXTERNAL REGISTER
        COMMON_REG;

    EXTERNAL ROUTINE
        ISSUE_IO : L$ISSUE_IO,                  ! Send an io to the tape drive
        GET_FIB : COMMON_CALL,          ! get user's file information block
        POSITION_TO_END : COMMON_CALL,          ! position volume set to end
        SPACE_IN_FILE   : COMMON_CALL,          ! space within file
        REWIND_FILE     : COMMON_CALL,          ! rewind file
        REWIND_VOL_SET  : COMMON_CALL;          ! rewind volume set

    EXTERNAL

        ! address of current unit control block
        !
        CURRENT_UCB     : REF BBLOCK,
        IO_PACKET       : REF BBLOCK,           ! address of current io request
                                                !   packet
        QUEUE_HEAD      : REF BBLOCK;           ! address of acp queue head

    LOCAL
        FIB             : REF BBLOCK,           ! address of copy of user's
```

```
2                                                ! file info block
2           FUNCTION        : BLOCK [1],         ! io function code and
2                                                !   modifiers
2           PACKET  : REF BBLOCK;                ! address of io request packet
2
2       PACKET = .IO_PACKET;                     ! get address  of io packet
2       FUNCTION = .PACKET[IRP$W_FUNC];     ! get function code and modifiers
2
2       IF .FUNCTION[IO$V_DMOUNT]
2           OR
2           .FUNCTION[IO$V_MOUNT]
2           OR
2           .FUNCTION[IO$V_REMOUNT]
2       THEN
2           RETURN;
2
2       IF NOT .PACKET[IRP$V_VIRTUAL]
2       THEN
3           BEGIN
3           KERNEL_CALL(CANCEL_IO);
3
4           IF (.CURRENT_VCB[VCB$V_WAIMOUVOL]
4               AND
4               NOT CANCEL_OP_REPLY())
3               OR
3               .CURRENT_VCB[VCB$V_WAIUSRLBL]
3           THEN
4               BEGIN
4               ERROR(SS$_CANCEL);
4               KERNEL_CALL(DO_CANCEL);
3               END;
3
3           ! Stall cancel until rewind or mount vol complete so cancels are not
3           ! continuously issued.
3           !
3
3           IF .CURRENT_VCB[VCB$V_WAIREWIND]
3               OR
3               .CURRENT_VCB[VCB$V_WAIMOUVOL]
3           THEN
3               KERNEL_CALL(STALL);
3
3           RETURN;
3
2           END;
2
2       FIB = GET_FIB(.BBLOCK[.PACKET[IRP$L_SVAPTE], AIB$L_DESCRIPT]);
2
2       IF .CURRENT_VCB[VCB$V_WAIUSRLBL]
2       THEN
2           ERR_EXIT(SS$_WAITUSRLBL);
2
2       IF .CURRENT_VCB[VCB$V_MUSTCLOSE]
2       THEN
2           ERR_EXIT(SS$_MUSTCLOSEFL);
2
2       ! Allow the user to clear the serious exception from the tape drive
```

```
2       ! by issuing a sensemode, which is effectively a NOP. This gives the
2       ! user the capibility to write blocks beyond EOT and before the
2       ! EOV labels. It also allows the user to read blocks beyond the
2       ! EOT and before the EOV labels. The user is never allowed to
2       ! read the EOV labels.
2       !
2       ! Please note that the case statement works on the assumption that
2       ! the variables are within a certain range. FIB$C_CLSEREXCP does
2       ! not fall in that range.
2
2       IF .FIB[FIB$W_CNTRLFUNC] EQL FIB$C_CLSEREXCP
2           THEN
3               BEGIN
3               ISSUE_IO(IO$_SENSEMODE, 0, 0);
3               RETURN;
2               END;
2
2       CASE .FIB[FIB$W_CNTRLFUNC] FROM FIB$C_REWINDVOL TO FIB$C_REWINDFIL OF
2           SET
2
2           [FIB$C_REWINDFIL] :
2               REWIND_FILE();
2
2           [FIB$C_POSEND] :
2               POSITION_TO_END();
2
2           [FIB$C_NEXTVOL] :
3               BEGIN
3
3               ! file must be accessed
3               !
3
3               IF .CURRENT_WCB EQL 0
3               THEN
3                   ERR_EXIT(SS$_FILNOTACC);
3
3               ! if not in data area, not appropriate time to be doing a next
3               ! volume
3               !
3
3               IF .CURRENT_VCB[VCB$B_TM] NEQ 1
3               THEN
3                   ERR_EXIT(SS$_ILLSEQOP);
3
3               KERNEL_CALL(STOP_VIO);
3
3               IF .CURRENT_WCB[WCB$V_READ]
3               THEN
4                   BEGIN                               ! read case
4                   SPACE_TM(1);                        ! space to trailer record
4
4                   IF NOT READ_BLOCK(.HDR1, ANSI_LBLSZ)
4                   THEN
4                       ERR_EXIT(SS$_TAPEPOSLOST);
4
4                   IF .HDR1[EO1$L_EO1LID] EQL 'EOF1'
4                   THEN
```

```
;  313          0695 4                       ERR_EXIT(SS$_ENDOFFILE);
;  314          0696 4
;  315          0697 4                   NEXT_VOL_READ();
;  316          0698 4                   END
;  317          0699 3               ELSE
;  318          0700 3                   NEXT_VOL_WRITE();                 ! write case
;  319          0701 3               KERNEL_CALL(START_VIO);               ! requeue blocked io
;  320          0702 2               END;
;  321          0703 2
;  322          0704 2           [FIB$C_SPACE] :
;  323          0705 2               SPACE_IN_FILE();
;  324          0706 2
;  325          0707 2           [FIB$C_REWINDVOL] :
;  326          0708 2               REWIND_VOL_SET();
;  327          0709 2
;  328          0710 2           [OUTRANGE] :
;  329          0711 2               ERR_EXIT(SS$_ILLCNTRFUNC);
;  330          0712 2
;  331          0713 2           [INRANGE] :
;  332          0714 2               ERR_EXIT(SS$_ILLCNTRFUNC);
;  333          0715 2           TES;
;  334          0716 2
;  335          0717 1       END;


                                                        .TITLE  ACPCTR
                                                        .IDENT  \V04-000\

                                                        .EXTRN  CANCEL_OP_REPLY
                                                        .EXTRN  IO_DONE, NEXT_VOL_READ
                                                        .EXTRN  NEXT_VOL_WRITE, READ_BLOCK
                                                        .EXTRN  RET_FREE_PAGE, RETURN_ALL_ERR
                                                        .EXTRN  SEND_ERRLOG, SPACE_TM
                                                        .EXTRN  START_VIO, STOP_VIO
                                                        .EXTRN  SYS$QIOW, ZERO_CHANNEL
                                                        .EXTRN  SCH$GL_PCBVEC, CURRENT_UCB
                                                        .EXTRN  CURRENT_WCB, HDR1
                                                        .EXTRN  IO_CHANNEL, IO_PACKET
                                                        .EXTRN  USER_STATUS, ISSUE_IO
                                                        .EXTRN  GET_FIB, POSITION_TO_END
                                                        .EXTRN  SPACE_IN_FILE, REWIND_FILE
                                                        .EXTRN  REWIND_VOL_SET, QUEUE_HEAD
                                                        .EXTRN  SYS$CMKRNL

                                                        .PSECT  $CODE$,NOWRT,2

                                    0000 00000          .ENTRY  MTA_ACPCNTRL, Save nothing                          : 0524
                  52      0000G   CF  D0 00002          MOVL    IO_PACKET, PACKET                                   : 0586
                  50         20   A2  3C 00007          MOVZWL  32(PACKET), FUNCTION                                : 0587
        01        50              0A  E1 0000B          BBC     #10, FUNCTION, 1$                                   : 0589
                                      04 0000F          RET                                                         :
        01        50              09  E1 00010 1$:      BBC     #9, FUNCTION, 2$                                    : 0591
                                      04 00014          RET                                                         :
        01        50              0B  E1 00015 2$:      BBC     #11, FUNCTION, 3$                                   : 0593
                                      04 00019          RET                                                         :
        4D        2A      A2      04  E0 0001A 3$:      BBS     #4, 42(PACKET), 8$                                  : 0597
                                  7E  D4 0001F          CLRL    -(SP)                                               : 0600
```

```
                                   5E  DD 00021         PUSHL   SP                                        ;
                     0000V  CF     9F 00023             PUSHAB  CANCEL_IO                                  ;
         00000000G   9F            03  FB 00027         CALLS   #3, @#SYS$CMKRNL                           ;
      08        0B   AB            02  E1 0002E         BBC     #2, 11(CURRENT_VCB), 4$                    ; 0602
            0000G   CF             00  FB 00033         CALLS   #0, CANCEL_OP_REPLY                        ; 0604
                    05             50  E9 00038         BLBC    R0, 5$                                     ;
      16        0B   AB            04  E1 0003B 4$:     BBC     #4, 11(CURRENT_VCB), 6$                    ; 0606
            0000G   CF       0830  8F  B0 00040 5$:     MOVW    #2096, USER_STATUS                         ; 0609
                                   7E  D4 00047         CLRL    -(SP)                                      ; 0610
                                   5E  DD 00049         PUSHL   SP                                         ;
                     0000V  CF     9F 0004B             PUSHAB  DO_CANCEL                                  ;
         00000000G   9F            03  FB 0004F         CALLS   #3, @#SYS$CMKRNL                           ;
      06        0B   AB            03  E0 00056 6$:     BBS     #3, 11(CURRENT_VCB), 7$                    ; 0617
      01        0B   AB            02  E0 0005B         BBS     #2, 11(CURRENT_VCB), 7$                    ; 0619
                                       04 00060         RET                                                ;
                                   7E  D4 00061 7$:     CLRL    -(SP)                                      ; 0621
                                   5E  DD 00063         PUSHL   SP                                         ;
                     0000V  CF     9F 00065             PUSHAB  STALL                                      ;
                            00B4   31 00069             BRW     22$                                        ;
                              2C   B2  DD 0006C 8$:     PUSHL   @44(PACKET)                                ; 0627
            0000G   CF             01  FB 0006F         CALLS   #1, GET_FIB                                ;
                    52             50  D0 00074         MOVL    R0, FIB                                    ;
      04        0B   AB            04  E1 00077         BBC     #4, 11(CURRENT_VCB), 9$                    ; 0629
                             0950  8F  BF 0007C         CHMU    #2384                                      ; 0631
      04        0B   AB            06  E1 00080 9$:     BBC     #6, 11(CURRENT_VCB), 10$                   ; 0633
                             0948  8F  BF 00085         CHMU    #2376                                      ; 0635
                    11       16    A2  B1 00089 10$:    CMPW    22(FIB), #17                               ; 0648
                                   0B  12 0008D         BNEQ    11$                                        ;
                                   7E  7C 0008F         CLRQ    -(SP)                                      ; 0651
                                   27  DD 00091         PUSHL   #39                                        ;
                            0000G  30 00093             BSBW    ISSUE_IO                                   ;
                    5E             0C  C0 00096         ADDL2   #12, SP                                    ;
                                       04 00099         RET                                                ; 0650
                05        01    16  A2  AF 0009A 11$:   CASEW   22(FIB), #1, #5                            ; 0655
    0089        001B      0015      008F    0009F 12$:  .WORD   24$-12$,-                                  ;
                          000F      0095    000A7               14$-12$,-                                  ;
                                                                15$-12$,-                                  ;
                                                                23$-12$,-                                  ;
                                                                25$-12$,-                                  ;
                                                                13$-12$                                    ;
                            0086   31 000AB             BRW     25$                                        ; 0711
            0000G   CF             00  FB 000AE 13$:    CALLS   #0, REWIND_FILE                            ; 0659
                                       04 000B3         RET                                                ;
            0000G   CF             00  FB 000B4 14$:    CALLS   #0, POSITION_TO_END                        ; 0662
                                       04 000B9         RET                                                ;
                             0000G  CF  D5 000BA 15$:   TSTL    CURRENT_WCB                                ; 0670
                                   04  12 000BE         BNEQ    16$                                        ;
                             00AC  8F  BF 000C0         CHMU    #172                                       ; 0672
                    01       2E    AB  91 000C4 16$:    CMPB    46(CURRENT_VCB), #1                        ; 0678
                                   04  13 000C8         BEQL    17$                                        ;
                             02DC  8F  BF 000CA         CHMU    #732                                       ; 0680
                                   7E  D4 000CE 17$:    CLRL    -(SP)                                      ; 0682
                                   5E  DD 000D0         PUSHL   SP                                         ;
                             0000G  CF  9F 000D2        PUSHAB  STOP_VIO                                   ;
         00000000G   9F            03  FB 000D6         CALLS   #3, @#SYS$CMKRNL                           ;
                    50       0000G  CF  D0 000DD        MOVL    CURRENT_WCB, R0                            ; 0684
                    2F       0B    A0  E9 000E2         BLBC    11(R0), 20$                                ;
```

```
                                          01  DD 000E6          PUSHL   #1                                          ; 0687
                    0000G  CF             01  FB 000E8          CALLS   #1, SPACE_TM                                ;
                           7E       50    8F  9A 000ED          MOVZBL  #80, -(SP)                                  ; 0689
                                 0000G    CF  DD 000F1          PUSHL   HDR1                                        ;
                    0000G  CF             02  FB 000F5          CALLS   #2, READ_BLOCK                              ;
                           04             50  E8 000FA          BLBS    R0, 18$                                     ;
                                  0224    8F  BF 000FD          CHMU    #548                                        ; 0691
                 31464F45  8F    0000G    DF  D1 00101   18$:   CMPL    @HDR1, #826691397                           ; 0693
                                          04  12 0010A          BNEQ    19$                                         ;
                                  0870    8F  BF 0010C          CHMU    #2160                                       ; 0695
                                      0000G   30 00110   19$:   BSBW    NEXT_VOL_READ                               ; 0697
                                          03  11 00113          BRB     21$                                         ; 0684
                                      0000G   30 00115   20$:   BSBW    NEXT_VOL_WRITE                              ; 0700
                                          7E  D4 00118   21$:   CLRL    -(SP)                                       ; 0701
                                          5E  DD 0011A          PUSHL   SP                                          ;
                                 0000G    CF  9F 0011C          PUSHAB  START_VIO                                   ;
                000000000G  9F            03  FB 00120   22$:   CALLS   #3, @#SYS$CMKRNL                            ;
                                          04  00127             RET                                                 ; 0655
                    0000G  CF             00  FB 00128   23$:   CALLS   #0, SPACE_IN_FILE                           ; 0705
                                          04  0012D             RET                                                 ;
                    0000G  CF             00  FB 0012E   24$:   CALLS   #0, REWIND_VOL_SET                          ; 0708
                                          04  00133             RET                                                 ;
                                  00E4    8F  BF 00134   25$:   CHMU    #228                                        ; 0714
                                          04  00138             RET                                                 ; 0717

; Routine Size:  313 bytes,    Routine Base:  $CODE$ + 0000


;  336          0718  1
```

```
ROUTINE MOUNT : COMMON_CALL NOVALUE =

!++
!
! FUNCTIONAL DESCRIPTION:
!       This routine gets a virtual page for the mounted volume to use
!
! CALLING SEQUENCE:
!       Mount(), must be called in kernel mode
!
! INPUT PARAMETERS:
!       None
!
! IMPLICIT INPUTS:
!       CURRENT_UCB - address of currnet unit control block
!       CURRENT_VCB - address of current volume control block
!
! OUTPUT PARAMETERS:
!       None
!
! IMPLICIT OUTPUTS:
!       Virtual page for volume to use
!
! ROUTINE VALUE:
!       None
!
! SIDE EFFECTS:
!       None
!
!--

    BEGIN

    EXTERNAL REGISTER
        COMMON_REG;

    EXTERNAL ROUTINE
        GET_FREE_PAGE   : COMMON_CALL;          ! get free virtual page

    EXTERNAL

        ! address of current unit control block
        !
        CURRENT_UCB     : REF BBLOCK;

    LOCAL
        JIB     : REF BBLOCK,
        PCB     : REF BBLOCK,
        VPAGE   : REF BBLOCK;   ! address of virtual page for volume set

    ! get virtual page for use by the volume set
    !
    GET_FREE_PAGE(1, VPAGE);
    VPAGE[VVP$B_TYPE] = VVP_TYPE;
    INSQUE(.VPAGE, CURRENT_VCB[VCB$L_VPFL]);
    VPAGE[VVP$L_STALLIOFL] = VPAGE[VVP$L_STALLIOFL];
    VPAGE[VVP$L_STALLIOBL] = VPAGE[VVP$L_STALLIOFL];
```

```
;  395          0776  2
;  396          0777  2      CURRENT_UCB[UCB$L_DEVCHAR] = .CURRENT_UCB[UCB$L_DEVCHAR]
;  397          0778  2                                    OR
;  398          0779  2                                    (DEV$M_MNT OR DEV$M_DIR OR DEV$M_SDI);
;  399          0780  2
;  400          0781  2      ! save the Account and User names
;  401          0782  2      !
;  402          0783  2      PCB = .SCH$GL_PCBVEC [ .(IO_PACKET[IRP$L_PID])<0, 16> ];
;  403          0784  2      JIB = .PCB [ PCB$L_JIB ];
;  404          0785  2      CH$MOVE ( VVP$S_USERNAME, JIB [JIB$T_USERNAME], VPAGE [VVP$T_USERNAME] );
;  405          0786  2      CH$MOVE ( VVP$S_ACCOUNT,  JIB [JIB$T_ACCOUNT],  VPAGE [VVP$T_ACCOUNT] );
;  406          0787  1      END;                                       ! end of page


                                                                     .EXTRN  GET_FREE_PAGE

                                                 00FC 00000 MOUNT:   .WORD   Save R2,R3,R4,R5,R6,R7                       ; 0719
                           5E                 04  C2 00002           SUBL2   #4, SP                                       ;
                                              5E  DD 00005           PUSHL   SP                                           ; 0771
                                              01  DD 00007           PUSHL   #1                                           ;
                     0000G CF                 02  FB 00009           CALLS   #2, GET_FREE_PAGE                            ;
                           50                 6E  D0 0000E           MOVL    VPAGE, R0                                    ; 0772
                        0A A0                 02  90 00011           MOVB    #2, 10(R0)                                   ;
                           51        3C       AB  9E 00015           MOVAB   60(R11), R1                                  ; 0773
                           61                 60  0E 00019           INSQUE  (R0), (R1)                                   ;
                           57                 6E  D0 0001C           MOVL    VPAGE, R7                                    ; 0774
                           50      01A4       C7  9E 0001F           MOVAB   420(R7), R0                                  ;
                           60                 50  D0 00024           MOVL    R0, (R0)                                     ;
                     01A8  C7                 50  D0 00027           MOVL    R0, 424(R7)                                  ; 0775
                           50     0000G       CF  D0 0002C           MOVL    CURRENT_UCB, R0                              ; 0777
                        38 A0  00080018       8F  C8 00031           BISL2   #524312, 56(R0)                              ; 0779
                           51  00000000G      9F  D0 00039           MOVL    @#SCH$GL_PCBVEC, R1                          ; 0783
                           50     0000G       CF  D0 00040           MOVL    IO_PACKET, R0                                ;
                           50                 0C  C0 00045           ADDL2   #12, R0                                      ;
                           50                 60  3C 00048           MOVZWL  (R0), R0                                     ;
                           50               6140  D0 0004B           MOVL    (R1)[R0], PCB                                ;
                           56      0080       C0  D0 0004F           MOVL    128(PCB), JIB                                ; 0784
          01B0  C7      0C A6                 0C  28 00054           MOVC3   #12, 12(JIB), 432(R7)                        ; 0785
          01BC  C7      18 A6                 08  28 0005B           MOVC3   #8, 24(JIB), 444(R7)                         ; 0786
                                                  04 00062           RET                                                  ; 0787

; Routine Size:  99 bytes,    Routine Base:  $CODE$ + 0139
;  408      0788  1 ROUTINE CANCEL_IO : COMMON_CALL NOVALUE =
;  409      0789  1
;  410      0790  1 !++
;  411      0791  1 !
;  412      0792  1 ! FUNCTIONAL DESCRIPTION:
;  413      0793  1 !       This routine sets the cancel io indicator if a file is accessed.
;  414      0794  1 !
;  415      0795  1 ! CALLING SEQUENCE:
;  416      0796  1 !       CANCEL_IO()
;  417      0797  1 !
;  418      0798  1 ! INPUT PARAMETERS:
;  419      0799  1 !       None
;  420      0800  1 !
;  421      0801  1 ! IMPLICIT INPUTS:
;  422      0802  1 !       CURRENT_VCB - address of current volume control block
;  423      0803  1 !
;  424      0804  1 ! OUTPUT PARAMETERS:
;  425      0805  1 !       None
;  426      0806  1 !
;  427      0807  1 ! IMPLICIT OUTPUTS:
;  428      0808  1 !       None
;  429      0809  1 !
;  430      0810  1 ! ROUTINE VALUE:
;  431      0811  1 !       None
;  432      0812  1 !
;  433      0813  1 ! SIDE EFFECTS:
;  434      0814  1 !       None
;  435      0815  1 !
;  436      0816  1 ! USER ERRORS:
;  437      0817  1 !       None
;  438      0818  1 !
;  439      0819  1 !--
;  440      0820  1
;  441      0821  2     BEGIN
;  442      0822  2
;  443      0823  2     EXTERNAL REGISTER
;  444      0824  2         COMMON_REG;
;  445      0825  2
;  446      0826  2     IF .CURRENT_VCB[VCB$L_WCB] NEQ 0
;  447      0827  2         OR
;  448      0828  2         .CURRENT_VCB[VCB$V_WAIREWIND]
;  449      0829  2         OR
;  450      0830  2         .CURRENT_VCB[VCB$V_WAIMOUVOL]
;  451      0831  2     THEN
;  452      0832  2
;  453      0833  2         ! remember that cancel was issued
;  454      0834  2         !
;  455      0835  2         CURRENT_VCB[VCB$V_CANCELIO] = 1;
;  456      0836  2
;  457      0837  1     END;



                                        0000 00000 CANCEL_IO:
                                                        .WORD   Save nothing                          ; 0788
```

```
                                          38   AB  D5 00002          TSTL    56(CURRENT_VCB)                        ; 0826
                                               0A  12 00005          BNEQ    1$                                     ;
                   05        0B    AB          03  E0 00007          BBS     #3, 11(CURRENT_VCB), 1$                ; 0828
                   04        0B    AB          02  E1 0000C          BBC     #2, 11(CURRENT_VCB), 2$                ; 0830
                             0B    AB          20  88 00011 1$:      BISB2   #32, 11(CURRENT_VCB)                   ; 0835
                                                   04 00015 2$:      RET                                            ; 0837

; Routine Size:  22 bytes,    Routine Base:  $CODE$ + 019C


;  458          0838  1
```

```
1 GLOBAL ROUTINE DO_CANCEL : COMMON_CALL NOVALUE =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !       This routine cancels all blocked io and acp functions.
1 !
1 ! CALLING SEQUENCE:
1 !       DO_CANCEL(), called in kernel mode
1 !
1 ! INPUT PARAMETERS:
1 !       None
1 !
1 ! IMPLICIT INPUTS:
1 !       CURRENT_VCB - address of current volume control block
1 !       USER_STATUS - contains error code which blocked io should be returned with
1 !
1 ! OUTPUT PARAMETERS:
1 !       USER_STATUS - reset to normal status
1 !
1 ! IMPLICIT OUTPUTS:
1 !       None
1 !
1 ! ROUTINE VALUE:
1 !       None
1 !
1 ! SIDE EFFECTS:
1 !       None
1 !
1 ! USER ERRORS:
1 !       None
1 !
1 !--
1
2     BEGIN
2
2     EXTERNAL REGISTER
2         COMMON_REG;
2
2     EXTERNAL ROUTINE
2         CHECK_DISMOUNT  : COMMON_CALL;          ! get free virtual page
2
2     MAP
2         USER_STATUS : LONG;
2
2     LOCAL
2         ABD     : REF BBLOCKVECTOR [, ABD$C_LENGTH],
2
2         ! address containing information about blocked request
2         !
2         BLOCK_PAGE,
2         FUNCTION        : BLOCK [1],
2         PACKET  : REF BBLOCK,           ! address of io blocked request
2         WINDOW;                         ! address of window for this request
2
2     ! If the process does not have a virtual page containing the information
2     ! describing the blocked request then have  fatal error
```

```
!

IF .CURRENT_VCB[VCB$L_VPFL] EQLA .CURRENT_VCB[VCB$L_VPBL]
THEN
    BUG_CHECK(NOBVPVCB);

REMQUE(.CURRENT_VCB[VCB$L_VPBL], BLOCK_PAGE);
PACKET = .(.BLOCK_PAGE + VVP$K_LENGTH + IO_PACKET - USER_STATUS);
RET_FREE_PAGE(.BLOCK_PAGE,FALSE);    ! return page(s) to virtual memory
RETURN_ALL_ERR();            ! return all blocked physical io in error

IF .CURRENT_VCB[VCB$V_WAIMOUVOL]
    OR
    .CURRENT_VCB[VCB$V_WAIREWIND]
THEN
    TERMINATE_VOL(.CURRENT_VCB[VCB$L_WCB]);

! If fib descriptor present, zero count so the fib is not returned.
! complete i/o.
!

IF .PACKET NEQ 0
THEN
    BEGIN

    !
    !

    IF .PACKET[IRP$V_COMPLX]
    THEN
        BEGIN
        FUNCTION = .PACKET[IRP$W_FUNC];
        ABD = .BBLOCK[.PACKET[IRP$L_SVAPTE], AIB$L_DESCRIPT];

        IF .FUNCTION[IO$V_ACCESS]
        THEN
            ZERO_CHANNEL(.PACKET)
        ELSE
            BEGIN
            FUNCTION = .PACKET[IRP$V_FCODE];

            IF .FUNCTION NEQ IO$_DEACCESS
            THEN
                ABD[ABD$C_WINDOW, ABD$W_COUNT] = 0;

            END;

        ABD[ABD$C_FIB, ABD$W_COUNT] = 0;
        END;

    IO_DONE(.PACKET);
    END;

! return stalled i/o with cancel
!

WHILE 1
```

```
2       DO
3           BEGIN
3
3           LOCAL
3               SAVE_STATUS;
3
3           IF REMQUE(.BBLOCK[.CURRENT_VCB[VCB$L_VPFL], VVP$L_STALLIOFL], PACKET)
3           THEN
3               EXITLOOP;
3
3           IF .PACKET[IRP$V_COMPLX]
3           THEN
4               BEGIN
4               FUNCTION = .PACKET[IRP$W_FUNC];
4               ABD = .BBLOCK[.PACKET[IRP$L_SVAPTE], AIB$L_DESCRIPT];
4
4               IF .FUNCTION[IO$V_ACCESS]
4               THEN
4                   ZERO_CHANNEL(.PACKET)
4               ELSE
4                   ABD[ABD$C_WINDOW, ABD$W_COUNT] = 0;
4
4               ABD[ABD$C_FIB, ABD$W_COUNT] = 0;
3               END;
3
3           ! If this is a cancel request, return is with normal status
3           !
3           SAVE_STATUS = .USER_STATUS;
3           FUNCTION = .PACKET[IRP$V_FCODE];
3
3           IF .FUNCTION EQL IO$_ACPCONTROL
3               AND
3               NOT .PACKET[IRP$V_VIRTUAL]
3           THEN
3               USER_STATUS = 1;
3
3           IO_DONE(.PACKET);
3           USER_STATUS = .SAVE_STATUS;
2           END;
2
2       ! If no file is accessed, turn off cancel I/O bit now.
2       !
2
2       IF .CURRENT_VCB[VCB$L_WCB] EQL 0
2       THEN
3           BEGIN
3           CURRENT_VCB [ VCB$V_CANCELIO ] = 0;
3
3           ! If while the cancel I/O was pending a dismount could have been issued
3           ! and refused waiting for cancel I/O to complete.  Check for dismount.
3           !
3           CHECK_DISMOUNT ( .BBLOCK [ .CURRENT_VCB[VCB$L_RVT], RVT$L_UCBLST ] );
2           END;
2
2       CURRENT_VCB[VCB$V_WAIREWIND] = 0;            ! no longer waiting
2       CURRENT_VCB[VCB$V_WAIUSRLBL] = 0;
2       CURRENT_VCB[VCB$V_WAIMOUVOL] = 0;
```

```
;  631         1010  2     ERROR(SS$_NORMAL);            ! cancel function should complete normally
;  632         1011  1     END;



                                                                   .EXTRN  CHECK_DISMOUNT, BUG$_NOBVPVCB

                                              007C 00000           .ENTRY  DO_CANCEL, Save R2,R3,R4,R5,R6                         ; 0839
                            56     0000G  CF    9E 00002           MOVAB   USER_STATUS, R6
                     40     AB       3C   AB    D1 00007           CMPL    60(CURRENT_VCB), 64(CURRENT_VCB)                       ; 0898
                                          04    12 0000C           BNEQ    1$
                                              FEFF 0000E           BUGW                                                           ; 0900
                                             0000* 00010           .WORD   <BUG$_NOBVPVCB!4>
                            50       40   BB    0F 00012  1$:      REMQUE  @64(CURRENT_VCB), BLOCK_PAGE                           ; 0902
                            52     0000GCF40    9E 00016           MOVAB   IO_PACKET+12[BLOCK_PAGE], R2                           ; 0903
                            51            66    9E 0001C           MOVAB   USER_STATUS, R1
                            52            51    C2 0001F           SUBL2   R1, R2
                            52            62    D0 00022           MOVL    (R2), PACKET
                                          7E    D4 00025           CLRL    -(SP)                                                  ; 0904
                                          50    DD 00027           PUSHL   BLOCK_PAGE
                     0000G  CF            02    FB 00029           CALLS   #2, RET_FREE_PAGE
                     0000G  CF            00    FB 0002E           CALLS   #0, RETURN_ALL_ERR                                     ; 0905
              05       0B   AB            02    E0 00033           BBS     #2, 11(CURRENT_VCB), 2$                                ; 0907
              08       0B   AB            03    E1 00038           BBC     #3, 11(CURRENT_VCB), 3$                                ; 0909
                                     38   AB    DD 0003D  2$:      PUSHL   56(CURRENT_VCB)                                        ; 0911
                     0000V  CF            01    FB 00040           CALLS   #1, TERMINATE_VOL
                                          52    D5 00045  3$:      TSTL    PACKET                                                 ; 0917
                                          32    13 00047           BEQL    7$
              26       2A   A2            03    E1 00049           BBC     #3, 42(PACKET), 6$                                     ; 0924
                            55       20   A2    3C 0004E           MOVZWL  32(PACKET), FUNCTION                                   ; 0927
                            53       2C   B2    D0 00052           MOVL    @44(PACKET), ABD                                       ; 0928
              09            55            06    E1 00056           BBC     #6, FUNCTION, 4$                                       ; 0930
                                          52    DD 0005A           PUSHL   PACKET                                                 ; 0932
                     0000G  CF            01    FB 0005C           CALLS   #1, ZERO_CHANNEL
                                          0E    11 00061           BRB     5$
55        20  A2            06            00    EF 00063  4$:      EXTZV   #0, #6, 32(PACKET), FUNCTION                           ; 0935
                            34            55    D1 00069           CMPL    FUNCTION, #52                                          ; 0937
                                          03    13 0006C           BEQL    5$
                                     02   A3    B4 0006E           CLRW    2(ABD)                                                 ; 0939
                                     0A   A3    B4 00071  5$:      CLRW    10(ABD)                                                ; 0943
                                          52    DD 00074  6$:      PUSHL   PACKET                                                 ; 0946
                     0000G  CF            01    FB 00076           CALLS   #1, IO_DONE
                            50       3C   AB    D0 0007B  7$:      MOVL    60(CURRENT_VCB), R0                                    ; 0959
                            52     01A4   D0    0F 0007F           REMQUE  @420(R0), PACKET
                                          42    1D 00084           BVS     12$
              18       2A   A2            03    E1 00086           BBC     #3, 42(PACKET), 10$                                    ; 0963
                            55       20   A2    3C 0008B           MOVZWL  32(PACKET), FUNCTION                                   ; 0966
                            53       2C   B2    D0 0008F           MOVL    @44(PACKET), ABD                                       ; 0967
              09            55            06    E1 00093           BBC     #6, FUNCTION, 8$                                       ; 0969
                                          52    DD 00097           PUSHL   PACKET                                                 ; 0971
                     0000G  CF            01    FB 00099           CALLS   #1, ZERO_CHANNEL
                                          03    11 0009E           BRB     9$
                                     02   A3    B4 000A0  8$:      CLRW    2(ABD)                                                 ; 0973
                                     0A   A3    B4 000A3  9$:      CLRW    10(ABD)                                                ; 0975
                            54            66    D0 000A6  10$:     MOVL    USER_STATUS, SAVE_STATUS                               ; 0980
55        20  A2            06            00    EF 000A9           EXTZV   #0, #6, 32(PACKET), FUNCTION                           ; 0981
                            38            55    D1 000AF           CMPL    FUNCTION, #56                                          ; 0983
```

```
                                           08  12 000B2        BNEQ    11$                                        ;
                03        2A  A2           04  E0 000B4        BBS     #4, 42(PACKET), 11$                        ; 0985
                              66           01  D0 000B9        MOVL    #1, USER_STATUS                            ; 0987
                                           52  DD 000BC 11$:   PUSHL   PACKET                                     ; 0989
                       0000G  CF           01  FB 000BE        CALLS   #1, IO_DONE                                ;
                              66           54  D0 000C3        MOVL    SAVE_STATUS, USER_STATUS                   ; 0990
                                           B3  11 000C6        BRB     7$                                         ; 0952
                                     38    AB  D5 000C8 12$:   TSTL    56(CURRENT_VCB)                            ; 0996
                                           10  12 000CB        BNEQ    13$                                        ;
                          0B  AB           20  8A 000CD        BICB2   #32, 11(CURRENT_VCB)                       ; 0999
                              50     20    AB  D0 000D1        MOVL    32(CURRENT_VCB), R0                        ; 1004
                                     44    A0  DD 000D5        PUSHL   68(R0)                                     ;
                       0000G  CF           01  FB 000D8        CALLS   #1, CHECK_DISMOUNT                         ;
                          0B  AB           1C  8A 000DD 13$:   BICB2   #28, 11(CURRENT_VCB)                       ; 1009
                              66           01  B0 000E1        MOVW    #1, USER_STATUS                            ; 1010
                                               04 000E4        RET                                                ; 1011

; Routine Size:  229 bytes,    Routine Base:  $CODE$ + 01B2


;  633          1012  1
```

```
1 GLOBAL ROUTINE TERMINATE_VOL (WINDOW) : COMMON_CALL NOVALUE =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !       This routine terminates a mount request.  If a file is open
1 !       then the user must close the file.  The write indicator is cleared so
1 !       that eof trailers are not written on deaccess.  The volume is marked
1 !       not mounted and the volume position is marked ambiguous.
1 !
1 ! CALLING SEQUENCE:
1 !       TERMINATE_MOUNT(WINDOW), called in kernel mode
1 !
1 ! INPUT PARAMETERS:
1 !       ARG1 - address of window for request
1 !
1 ! IMPLICIT INPUTS:
1 !       None
1 !
1 ! OUTPUT PARAMETERS:
1 !       None
1 !
1 ! IMPLICIT OUTPUTS:
1 !       None
1 !
1 ! ROUTINE VALUE:
1 !       None
1 !
1 ! SIDE EFFECTS:
1 !       None
1 !
1 ! USER ERRORS:
1 !       None
1 !
1 !--
1
2     BEGIN
2
2     EXTERNAL ROUTINE
2         GET_CCB;
2
2     EXTERNAL REGISTER
2         COMMON_REG;
2
2     MAP
2         WINDOW  : REF BBLOCK;           ! address of window control block
2
2     LOCAL
2         MVL_ENTRY       : REF BBLOCK;                   ! address of MVL entry
2
2     IF .WINDOW NEQ 0
2     THEN                                        ! a file is open
3         BEGIN
3         CURRENT_VCB[VCB$V_NOWRITE] = 1;
3         CURRENT_VCB[VCB$V_MUSTCLOSE] = 1;       ! the file must be closed
2         END;
2
```

```
;  692          1070  2     IF .CURRENT_VCB[VCB$V_WAIMOUVOL]
;  693          1071  2     THEN
;  694          1072  3         BEGIN
;  695          1073  3
;  696          1074  3         LOCAL
;  697          1075  3             CCB : REF BBLOCK,
;  698          1076  3             UCB : REF VECTOR;
;  699          1077  3
;  700          1078  3         MVL_ENTRY = .CURRENT_VCB[VCB$L_MVL] + MVL$K_FIXLEN +
;  701          1079  3             ((.CURRENT_VCB[VCB$B_CUR_RVN] - 1)*MVL$K_LENGTH);
;  702          1080  3         MVL_ENTRY[MVL$V_MOUNTED] = 0;               ! volume is not mounted
;  703          1081  3         UCB = BBLOCK[.CURRENT_VCB[VCB$L_RVT], RVT$L_UCBLST];
;  704          1082  3         UCB = .UCB[.CURRENT_VCB[VCB$W_RVN]];
;  705          1083  3         CCB = GET_CCB ( .IO_CHANNEL );
;  706          1084  3         CCB[CCB$L_UCB] = .UCB;
;  707          1085  3         SYS$QIOW(0, .IO_CHANNEL,
;  708          1086  3             IO$_REWINDOFF
;  709          1087  3             OR
;  710          1088  3             IO$M_NOWAIT
;  711          1089  3             OR
;  712          1090  3             IO$M_CLSEREXCP, 0, 0, 0, 0, 0, 0, 0, 0, 0);
;  713          1091  3         SEND_ERRLOG(0, .UCB);
;  714          1092  3         CURRENT_VCB[VCB$B_CUR_RVN] = 0;         ! no volume is current
;  715          1093  3
;  716          1094  3         ! no file is current, ie: start at beginning
;  717          1095  3         !
;  718          1096  3         CURRENT_VCB[VCB$L_CUR_FID] = 0;
;  719          1097  2         END;
;  720          1098  2
;  721          1099  1     END;                                    ! end of routine TERMINATE_MOUNT


                                                           .EXTRN  GET_CCB

                                         0004 00000        .ENTRY  TERMINATE_VOL, Save R2                              : 1013
                           04    AC   D5 00002             TSTL    WINDOW                                              : 1063
                                 05   13 00005             BEQL    1$                                                  :
               0B    AB    C0    8F   88 00007             BISB2   #192, 11(CURRENT_VCB)                               : 1067
      58       0B    AB          02   E1 0000C 1$:         BBC     #2, 11(CURRENT_VCB), 2$                             : 1070
                     50    2F    AB   9A 00011             MOVZBL  47(CURRENT_VCB), R0                                 : 1079
                     50    34  BB40   7E 00015             MOVAQ   @52(CURRENT_VCB)[R0], MVL_ENTRY                     : 1078
                     50          1C   C0 0001A             ADDL2   #28, MVL_ENTRY                                      :
               07    A0          01   8A 0001D             BICB2   #1, 7(MVL_ENTRY)                                    : 1080
      52       20    AB 00000044 8F   C1 00021             ADDL3   #68, 32(CURRENT_VCB), UCB                           : 1081
                     50    0E    AB   3C 0002A             MOVZWL  14(CURRENT_VCB), R0                                 : 1082
                     52        6240   D0 0002E             MOVL    (UCB)[R0], UCB                                      :
                        0000G    CF   DD 00032             PUSHL   IO_CHANNEL                                          : 1083
            0000G    CF          01   FB 00036             CALLS   #1, GET_CCB                                         :
                     60          52   D0 0003B             MOVL    UCB, (CCB)                                          : 1084
                                 7E   7C 0003E             CLRQ    -(SP)                                               : 1085
                                 7E   7C 00040             CLRQ    -(SP)                                               :
                                 7E   7C 00042             CLRQ    -(SP)                                               :
                                 7E   7C 00044             CLRQ    -(SP)                                               :
                                 7E   D4 00046             CLRL    -(SP)                                               :
                     7E     02A2 8F   3C 00048             MOVZWL  #674, -(SP)                                         : 1089
                        0000G    CF   DD 0004D             PUSHL   IO_CHANNEL                                          : 1085
```

```
                                          7E  D4 00051           CLRL     -(SP)                                         ;
                      00000000G  9F       0C  FB 00053           CALLS    #12, @#SYS$QIOW                               ;
                                          52  DD 0005A           PUSHL    UCB                                           ; 1091
                                          7E  D4 0005C           CLRL     -(SP)                                         ;
                          0000G  CF       02  FB 0005E           CALLS    #2, SEND_ERRLOG                               ;
                                     2F   AB  94 00063           CLRB     47(CURRENT_VCB)                               ; 1092
                                     24   AB  D4 00066           CLRL     36(CURRENT_VCB)                               ; 1096
                                              04 00069 2$:       RET                                                    ; 1099

; Routine Size:  106 bytes,    Routine Base:  $CODE$ + 0297

;  722          1100  1
;  724          1101  1 GLOBAL ROUTINE MTA_MOUNT : NOPRES NOVALUE =
;  725          1102  1
;  726          1103  1 !++
;  727          1104  1 !
;  728          1105  1 ! FUNCTIONAL DESCRIPTION:
;  729          1106  1 !       This routine checks the validity of the mount request and
;  730          1107  1 !       sets up a virtual page for this volume set.
;  731          1108  1 !
;  732          1109  1 !
;  733          1110  1 ! CALLING SEQUENCE:
;  734          1111  1 !       MTA_MOUNT()
;  735          1112  1 !
;  736          1113  1 ! INPUT PARAMETERS:
;  737          1114  1 !       None
;  738          1115  1 !
;  739          1116  1 ! IMPLICIT INPUTS:
;  740          1117  1 !       CURRENT_UCB     - address of current unit control block
;  741          1118  1 !       QUEUE_HEAD      - address of queue head for ACP
;  742          1119  1 !
;  743          1120  1 ! OUTPUT PARAMETERS:
;  744          1121  1 !       None
;  745          1122  1 !
;  746          1123  1 ! IMPLICIT OUTPUTS:
;  747          1124  1 !       one page of virtual memory is devoted to this volume set
;  748          1125  1 !
;  749          1126  1 ! ROUTINE VALUE:
;  750          1127  1 !       None
;  751          1128  1 !
;  752          1129  1 ! SIDE EFFECTS:
;  753          1130  1 !       None
;  754          1131  1 !
;  755          1132  1 !--
;  756          1133  1
;  757          1134  2     BEGIN
;  758          1135  2
;  759          1136  2     EXTERNAL
;  760          1137  2         CURRENT_UCB     : REF BBLOCK,
;  761          1138  2         QUEUE_HEAD      : REF BBLOCK;
;  762          1139  2
;  763          1140  2     EXTERNAL REGISTER
;  764          1141  2         COMMON_REG;
;  765          1142  2
;  766          1143  2     IF NOT .BBLOCK[CURRENT_UCB[UCB$L_DEVCHAR], DEV$V_SQD]
;  767          1144  2         OR
;  768          1145  2         .QUEUE_HEAD[AQB$B_ACPTYPE] NEQ AQB$K_MTA
;  769          1146  2     THEN
;  770          1147  2         ERR_EXIT(SS$_WRONGACP);
;  771          1148  2
;  772          1149  2     KERNEL_CALL(MOUNT);
;  773          1150  1     END;                                    ! end of routine MTA_MOUNT



                                     0000 00000          .ENTRY  MTA_MOUNT, Save nothing                         ; 1101
                        50     0000G  CF  D0 00002          MOVL    CURRENT_UCB, R0                                 ; 1143
```

```
          0B       38  A0             05  E1 00007          BBC     #5, 56(R0), 1$                         :
                       50     0000G   CF  D0 0000C          MOVL    QUEUE_HEAD, R0                         : 1145
                       03        15   A0  91 00011          CMPB    21(R0), #3                             :
                                      04  13 00015          BEQL    2$                                     :
                              031C    8F  BF 00017 1$:      CHMU    #796                                   : 1147
                                      7E  D4 0001B 2$:      CLRL    -(SP)                                  : 1149
                                      5E  DD 0001D          PUSHL   SP                                     :
                              FE15    CF  9F 0001F          PUSHAB  MOUNT                                  :
               00000000G  9F          03  FB 00023          CALLS   #3, @#SYS$CMKRNL                       :
                                      04  0002A             RET                                            : 1150

; Routine Size:  43 bytes,    Routine Base:  $CODE$ + 0301


;  774          1151  1
;  776         1152  1  ROUTINE STALL : COMMON_CALL NOVALUE =
;  777         1153  1
;  778         1154  1  !++
;  779         1155  1  !
;  780         1156  1  ! FUNCTIONAL DESCRIPTION:
;  781         1157  1  !
;  782         1158  1  !       This routine puts the cancel request packet on the stalled queue.
;  783         1159  1  !
;  784         1160  1  ! CALLING SEQUENCE:
;  785         1161  1  !       STALL(), called in KERNEL mode
;  786         1162  1  !
;  787         1163  1  ! INPUT PARAMETERS:
;  788         1164  1  !       None
;  789         1165  1  !
;  790         1166  1  ! IMPLICIT INPUTS:
;  791         1167  1  !       None
;  792         1168  1  !
;  793         1169  1  ! OUTPUT PARAMETERS:
;  794         1170  1  !       None
;  795         1171  1  !
;  796         1172  1  ! IMPLICIT OUTPUTS:
;  797         1173  1  !       cancel request queued to stall I/O queue
;  798         1174  1  !
;  799         1175  1  ! ROUTINE VALUE:
;  800         1176  1  !       None
;  801         1177  1  !
;  802         1178  1  ! SIDE EFFECTS:
;  803         1179  1  !       None
;  804         1180  1  !
;  805         1181  1  !--
;  806         1182  1
;  807         1183  2      BEGIN
;  808         1184  2
;  809         1185  2      EXTERNAL
;  810         1186  2          IO_PACKET       : REF BBLOCK;           ! address of current I/O packet
;  811         1187  2
;  812         1188  2      EXTERNAL REGISTER
;  813         1189  2          COMMON_REG;
;  814         1190  2
;  815         1191  2      LOCAL
;  816         1192  2          VPAGE   : REF BBLOCK;
;  817         1193  2
;  818         1194  2      VPAGE = .CURRENT_VCB[VCB$L_VPFL];
;  819         1195  2      INSQUE(.IO_PACKET, .VPAGE[VVP$L_STALLIOBL]);
;  820         1196  2      IO_PACKET = 0;
;  821         1197  1      END;



                                      0000 00000 STALL:   .WORD    Save nothing                                  ; 1152
                  50       3C    AB   D0 00002            MOVL     60(CURRENT_VCB), VPAGE                        ; 1194
          01A8    D0    0000G    DF   0E 00006            INSQUE   @IO_PACKET, @424(VPAGE)                       ; 1195
                        0000G    CF   D4 0000D            CLRL     IO_PACKET                                     ; 1196
                                      04 00011            RET                                                    ; 1197
```

```
; Routine Size:  18 bytes,    Routine Base:  $CODE$ + 032C

;  822          1198  1 END
;  823          1199  1
;  824          1200  0 ELUDOM



;                               PSECT SUMMARY
;
;       Name                        Bytes                       Attributes
;
;  $CODE$                               830  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)


;                           Library Statistics
;
;                                           -------- Symbols --------       Pages       Processing
;       File                                Total     Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[SYSLIB]LIB.L32;1            18619        74         0       1000          00:01.9



;                               COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:ACPCTR/OBJ=OBJ$:ACPCTR MSRC$:ACPCTR/UPDATE=(ENH$:ACPCTR)

; Size:          830 code + 0 data bytes
; Run Time:         00:20.6
; Elapsed Time:     00:56.8
; Lines/CPU Min:      3490
; Lexemes/CPU-Min: 19902
; Memory Used:  158 pages
; Compilation Complete
```

0253 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY